Chair for Cognitive Systems
Department of Electrical and Computer Engineering
Technische Universität München

Multi-Sensory Based Robot Dynamic Manipulation: Final project

Technical University of Munich
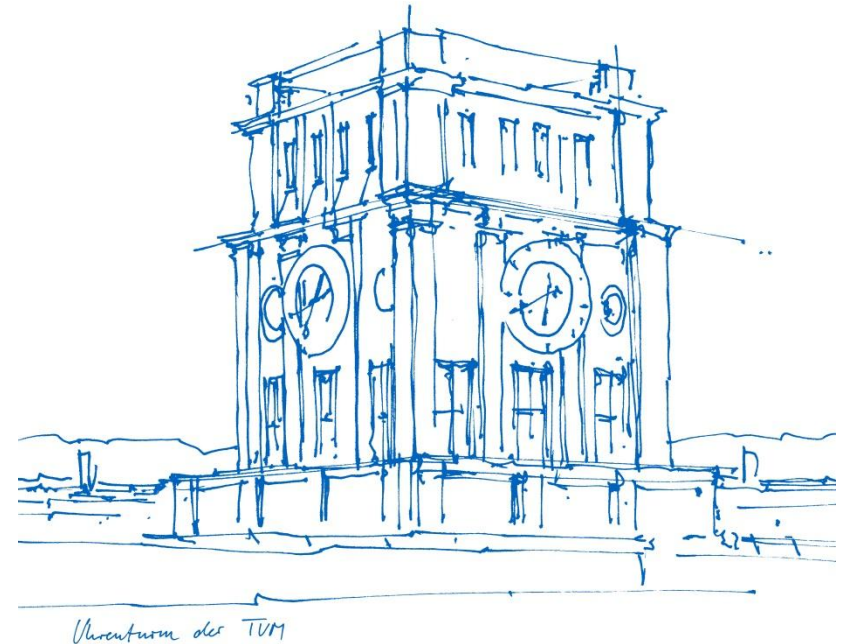
Chair for Cognitive Systems

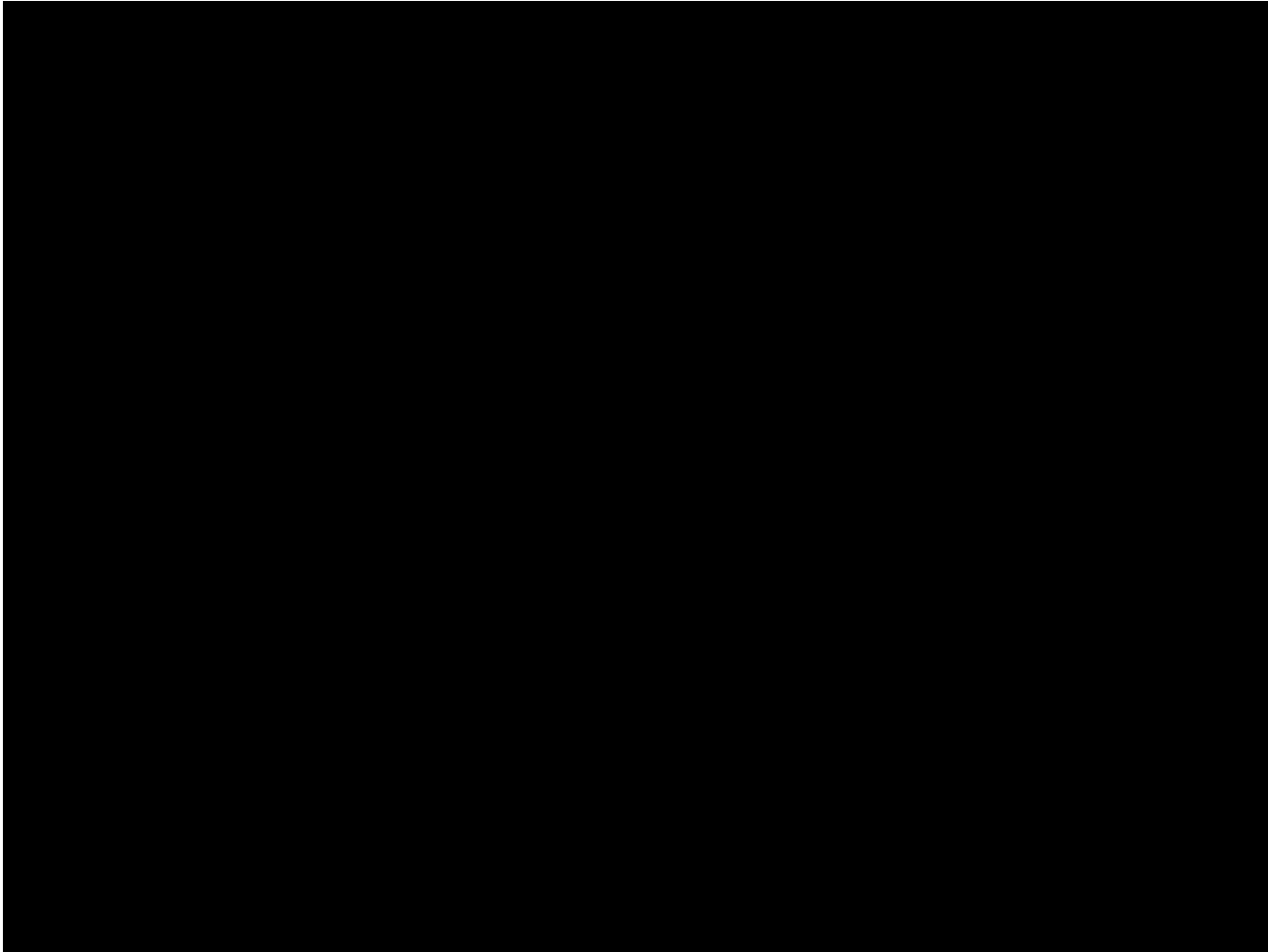Prof. Dr. Gordon Cheng

08. March 2022

Student Name:
Gintautas Palinauskas

Advisor:
M.Sc Simon Armleder

Uhrenturm der TUM

Demonstration Video

Control Decisions

- Orientation representation: quaternions.
- First positioning: joint control.
  - consider Qd, Qdp, Qdpp from polynomial interpolation (PI).
- Second positioning: operational control with impedance.
  - Circular motion: consider Xd from the trajectory generator.
  - Repositioning: consider Xd, Xdp linear part, Xdpp linear part from PI and slerp.
- Calculate tracking orientation using Euler angles.

Hyperparameters

Joint control:

p: [300,300,300,100,100,100]

d: [30,30,30,5,5,5]

i: [1,1,1,1,1,1]

gamma: 0.001

Operational control:

p: [20,20,20,15,15,15]

d: [40,40,40,25,25,25]

i: [0.01,0.01,0.01,0.01,0.01,0.01]

gamma: 1

Impedance control linear:

p: [100,100,100]

d: [300,300,300]

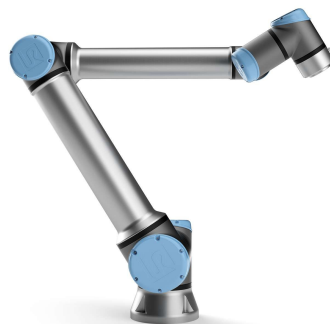Impedance control angular:

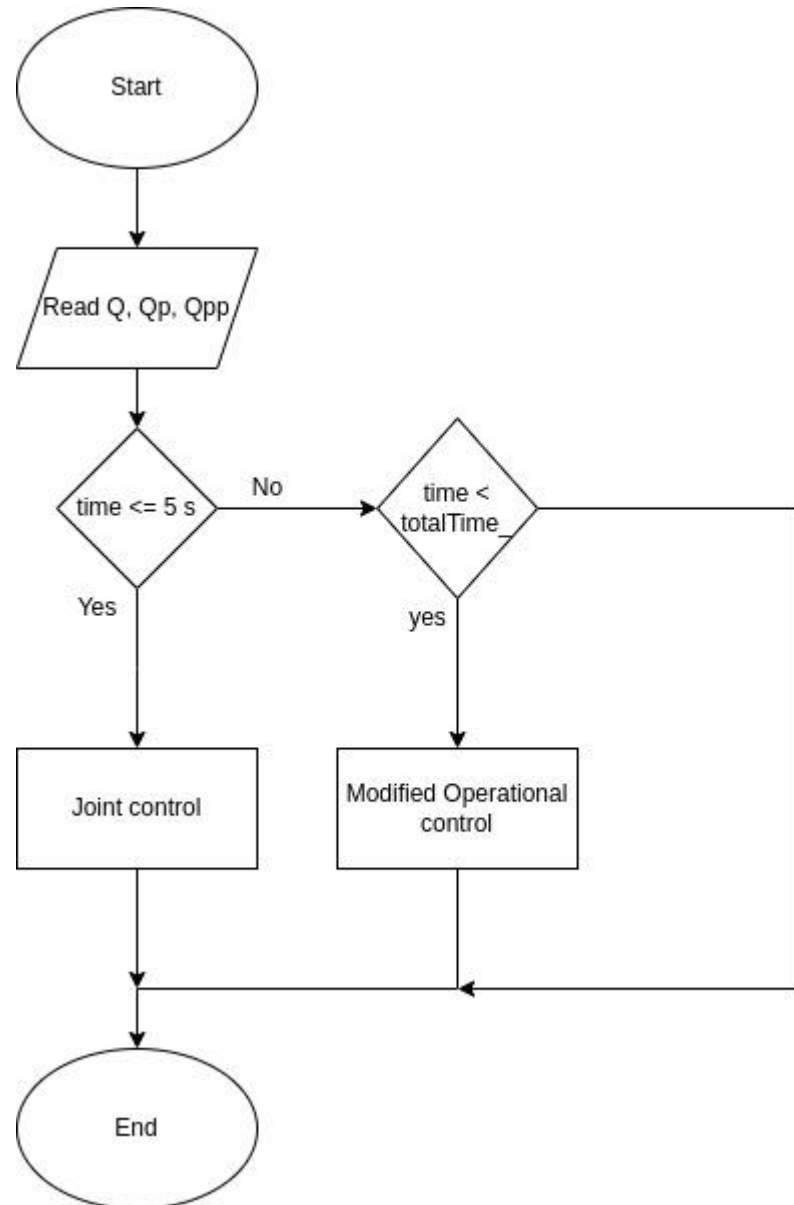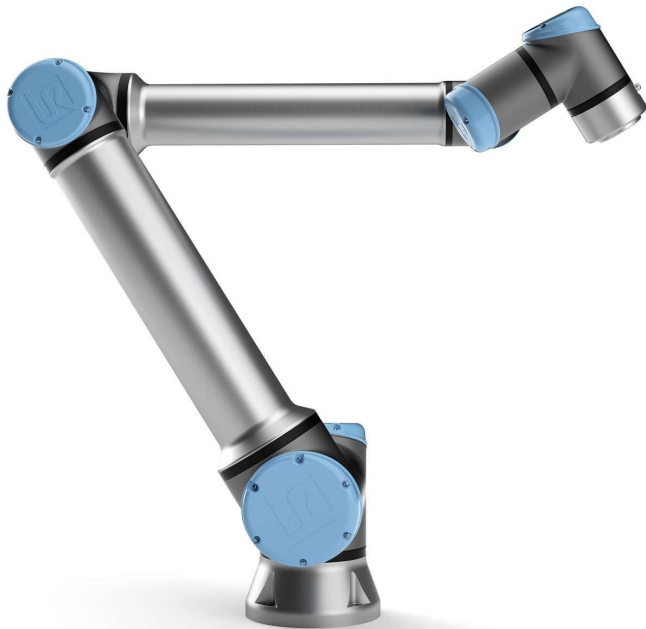p: [50,50,50]

d: [20,20,20]

influence radius: 0.3
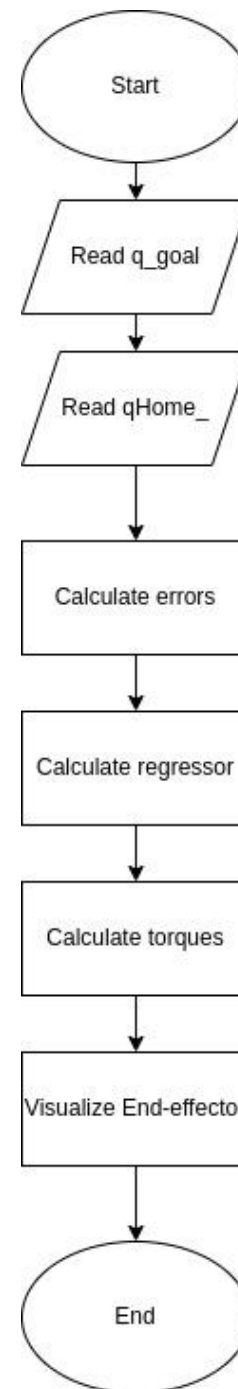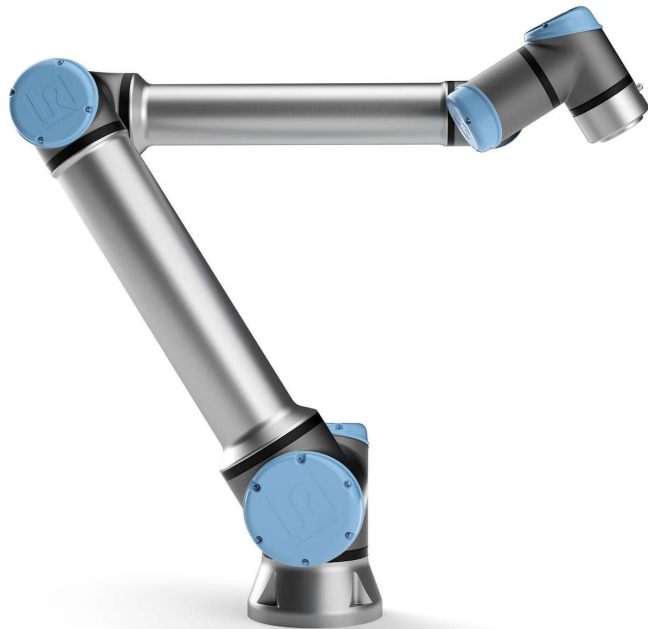
multiplier factor: 20

repelling force function: 1/x

time step: 0.002

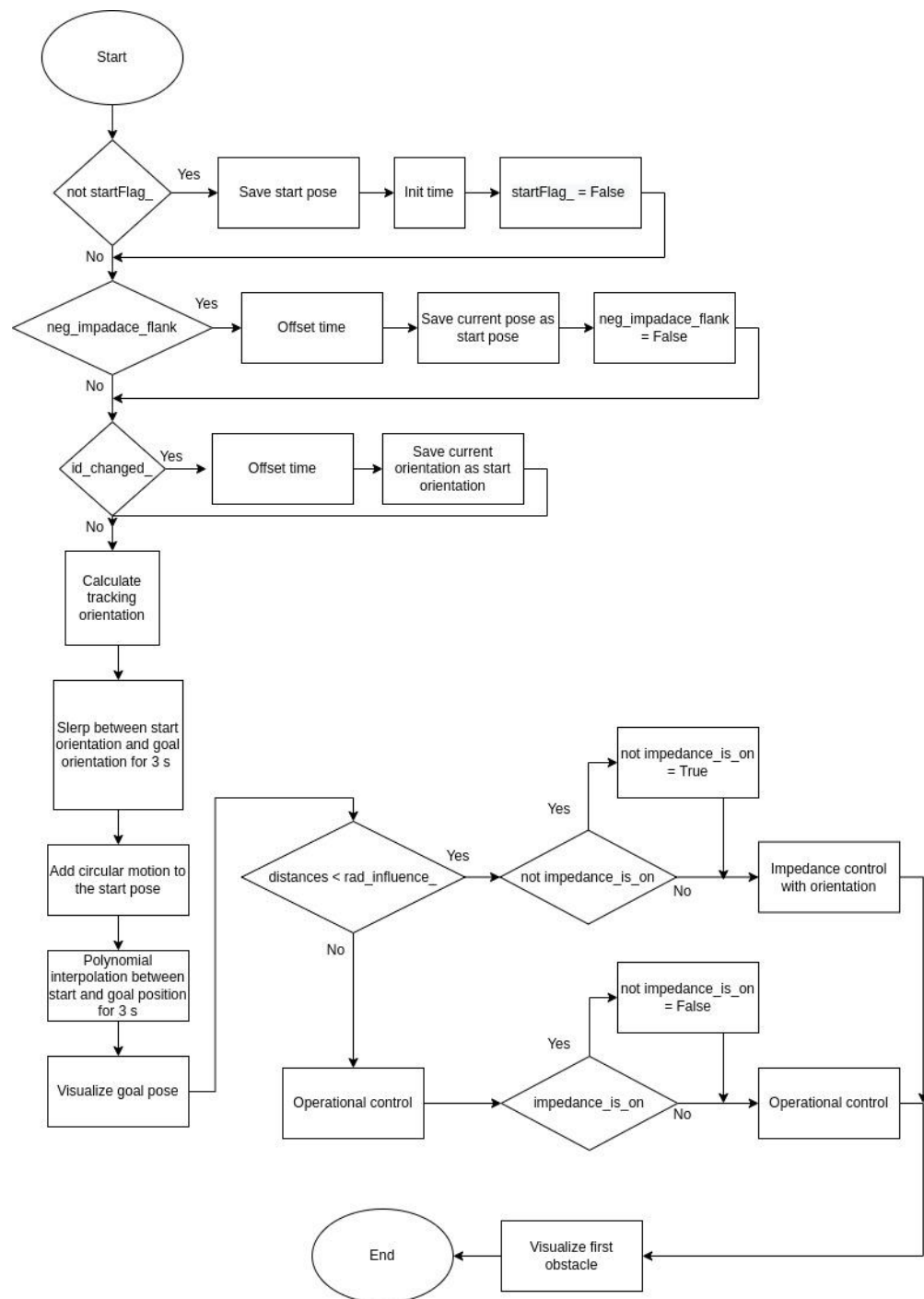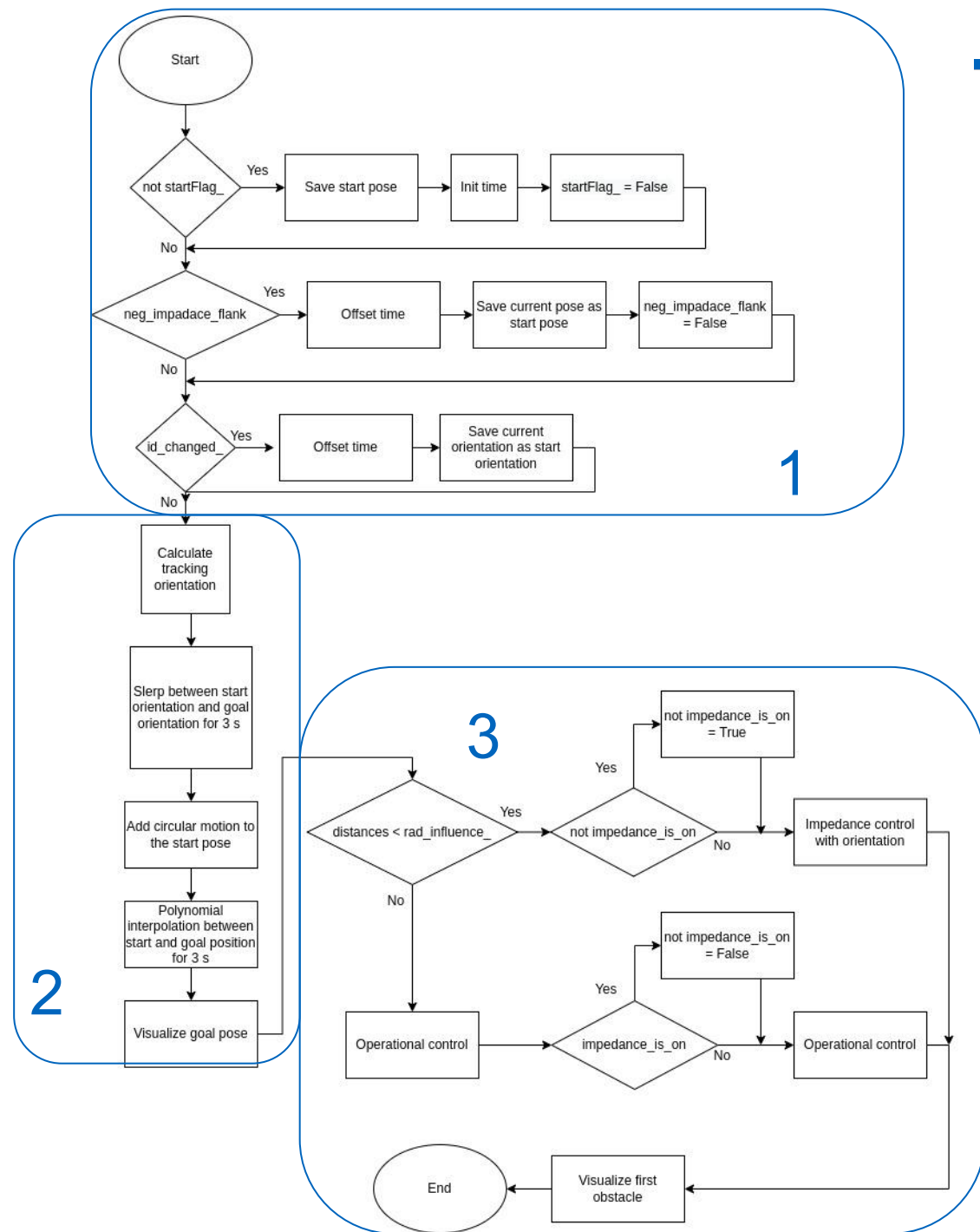# Update Flowchart

Joint Control Flowchart

Start → Read q_goal → Read qHome_ → Calculate errors → Calculate regressor → Calculate torques → Visualize End-effector → End
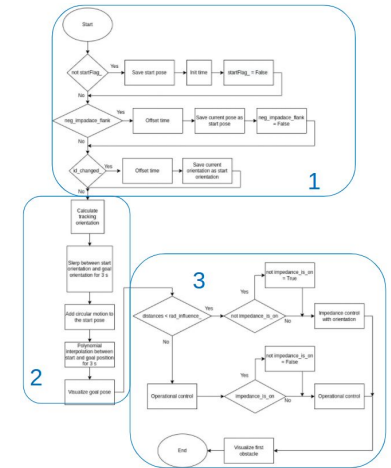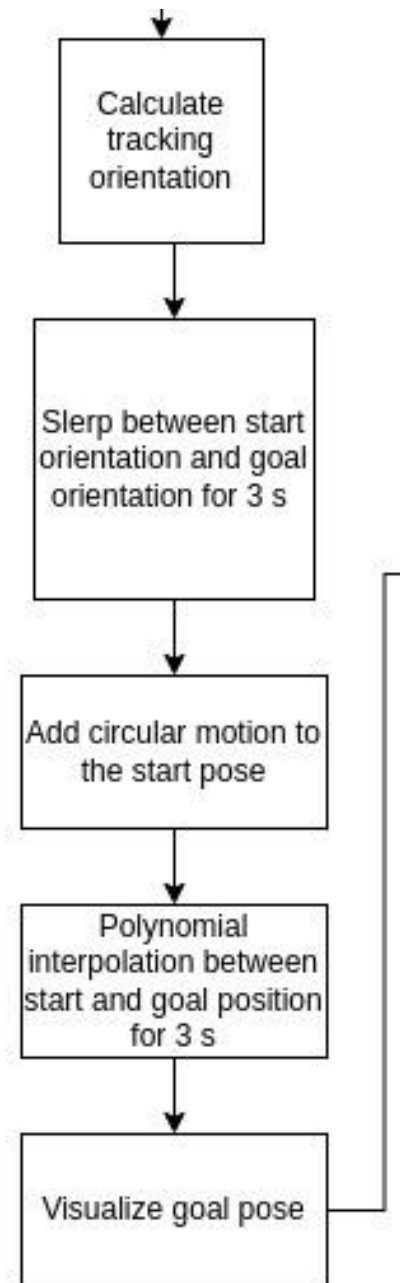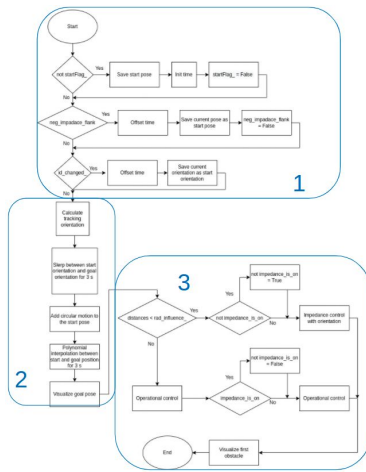
Modified Operational Control Flowchart
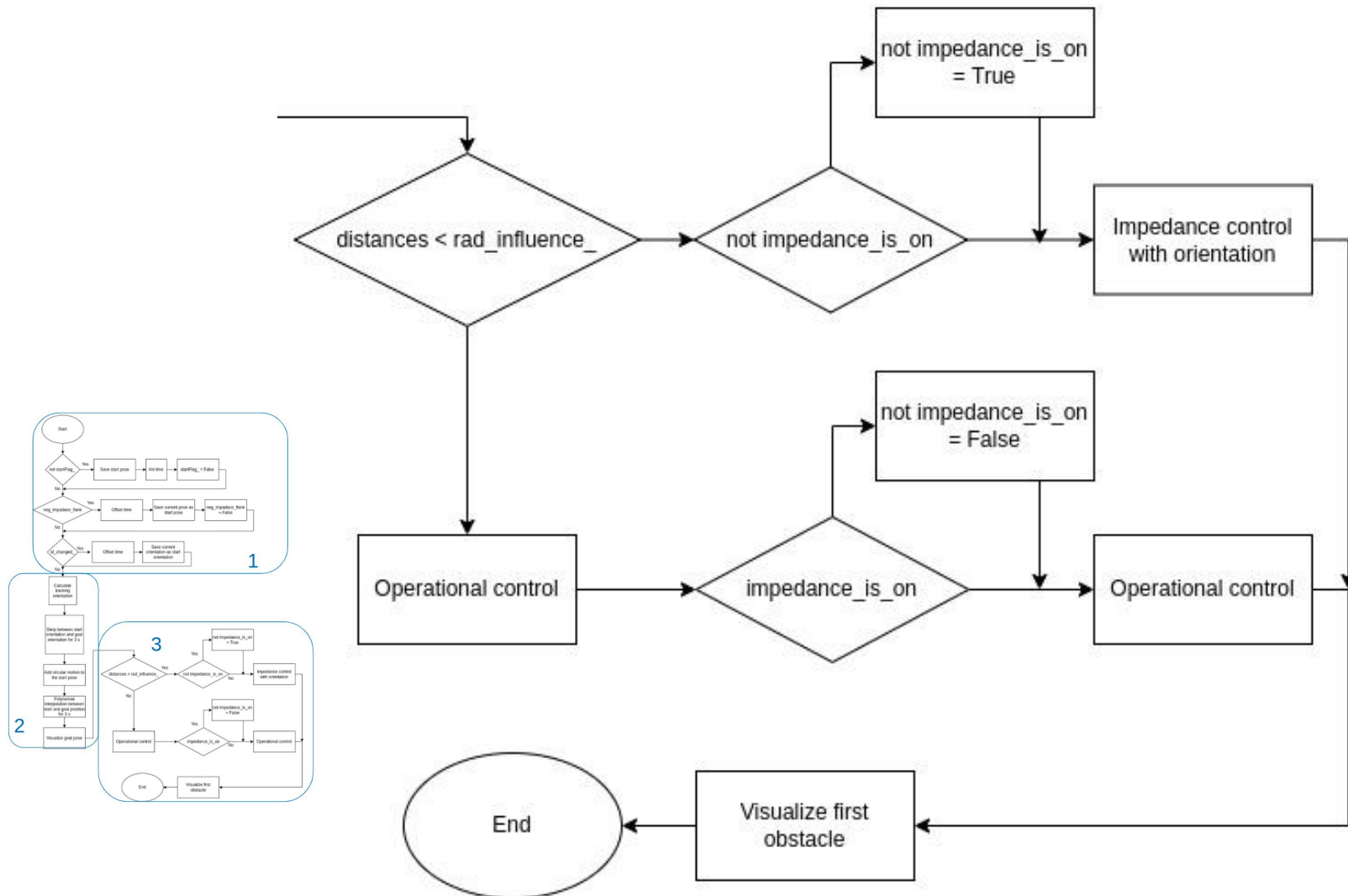
Modified Operational Control Flowchart

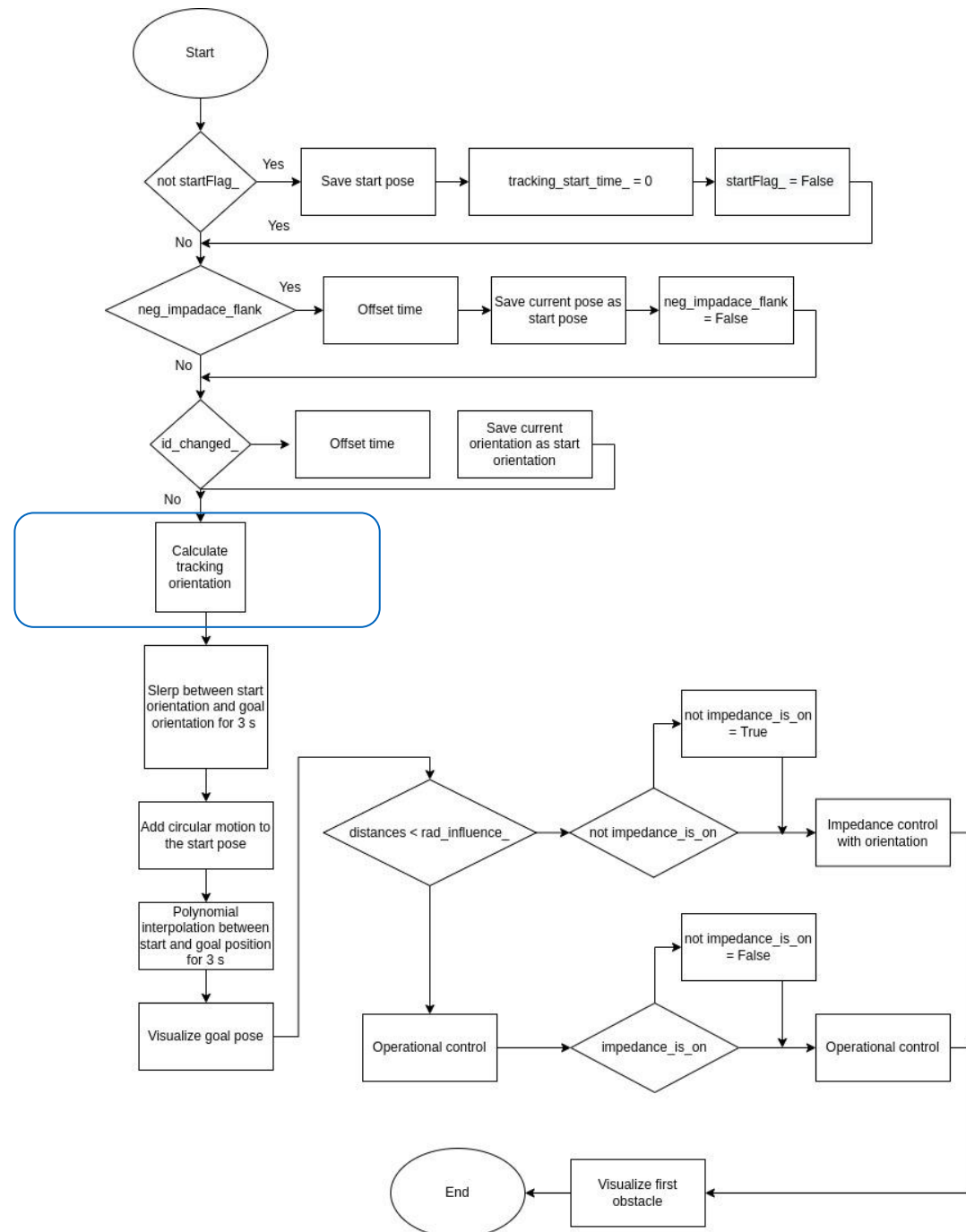# Modified Operational Control Flowchart. 1

Modified Operational Control Flowchart. 2

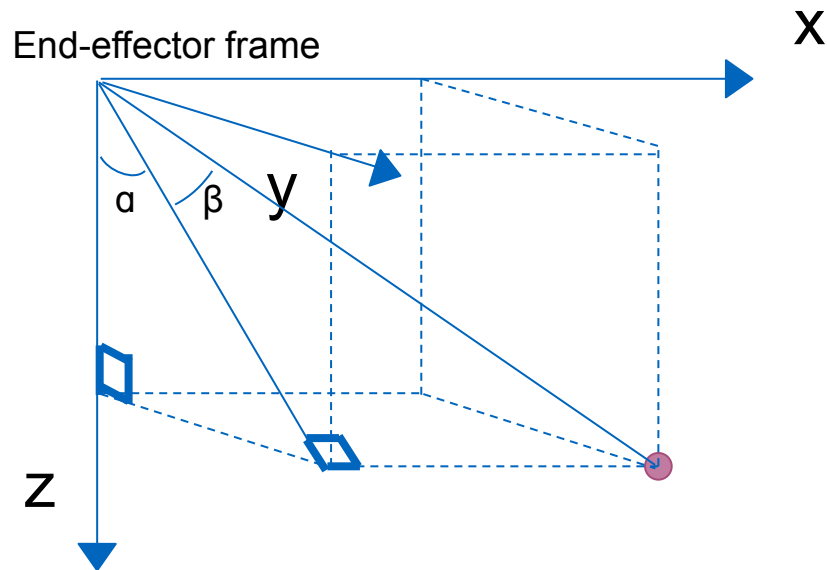# Modified Operational Control Flowchart. 3

Modified Operational Control Flowchart

Calculating Tracking Orientation

End-effector frame

x

α β y

z

Start

Transform target pose
to End-effector frame

Visualize target with a
frame

Calculate Euler
angles

Calculate goal
orientation

Transform End-
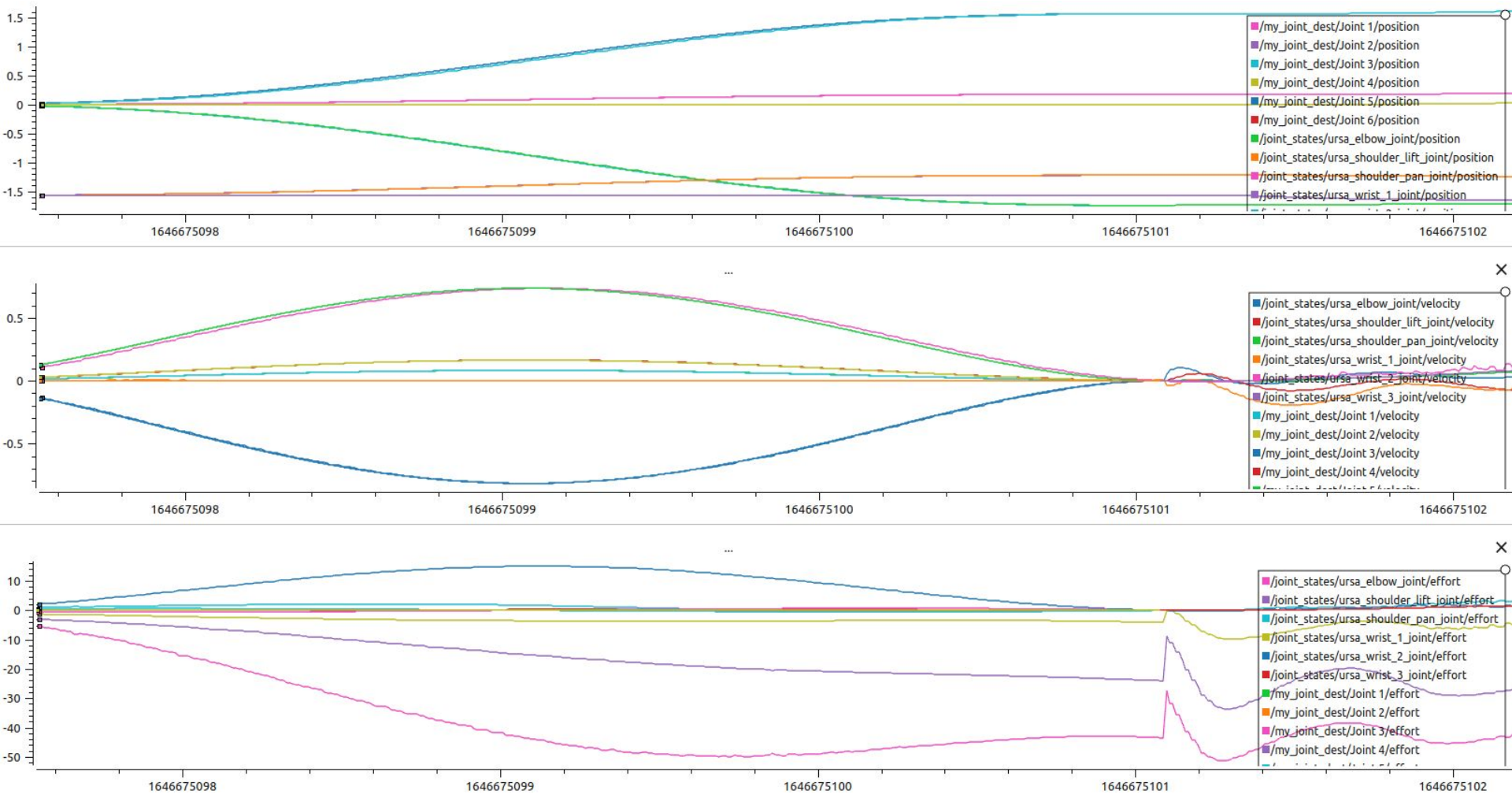effector orientation to
the goal orientation

End

```
Matrix3d R_goal =  R_ef_0 * Rx * Ry * Rz;
```
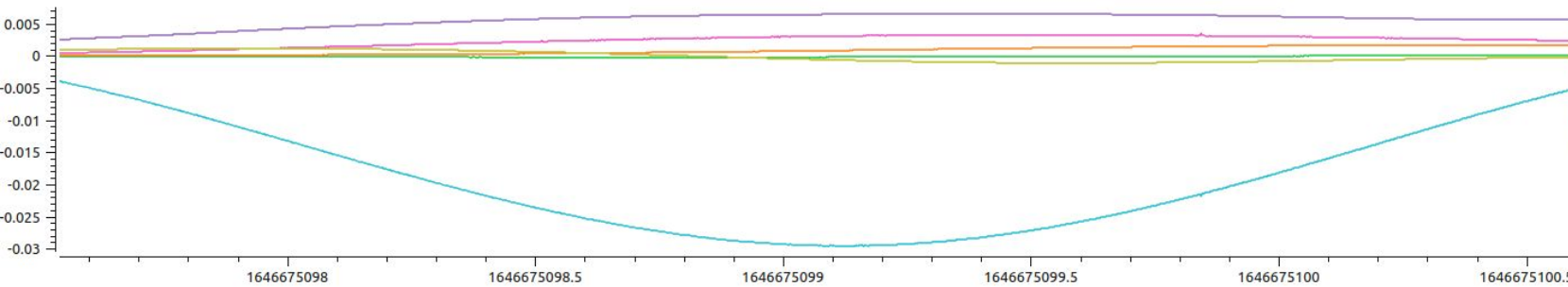
```
// Angle offsets for XYZ euler rotation
    double alpha = atan2(target_pos_ef(1), target_pos_ef(2));
    double beta = atan2(target_pos_ef(0), sqrt(pow(target_pos_ef(1),2) + pow(target_pos_ef(2),2)));
```
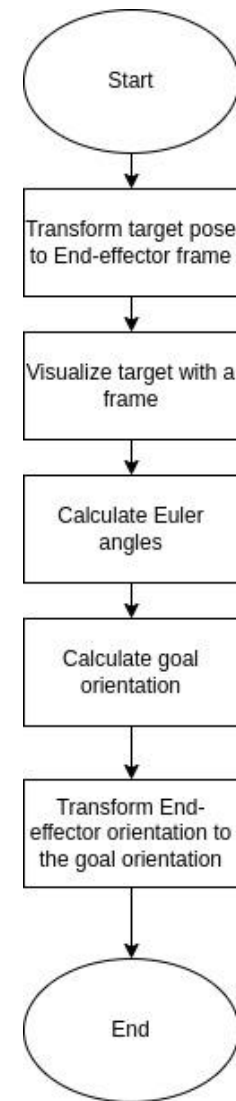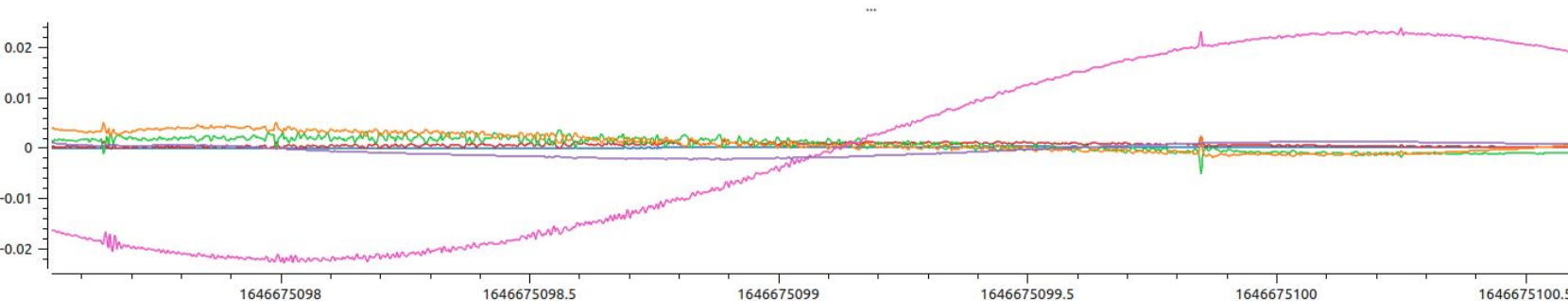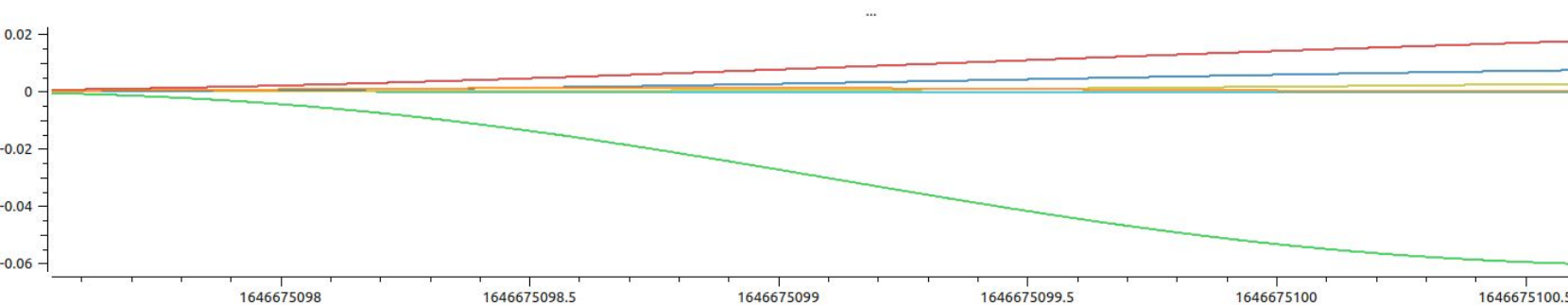
Joint Control Positioning Plots
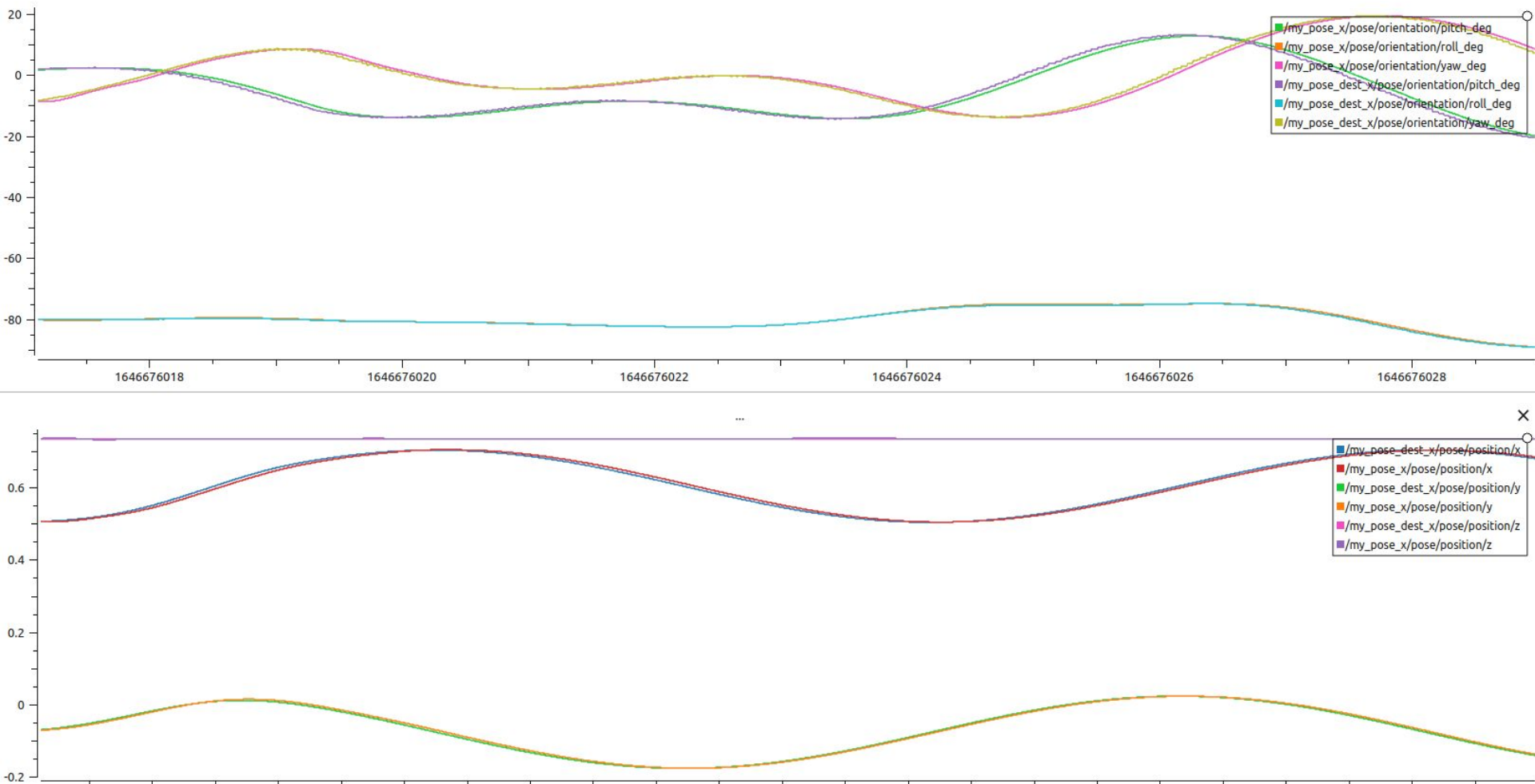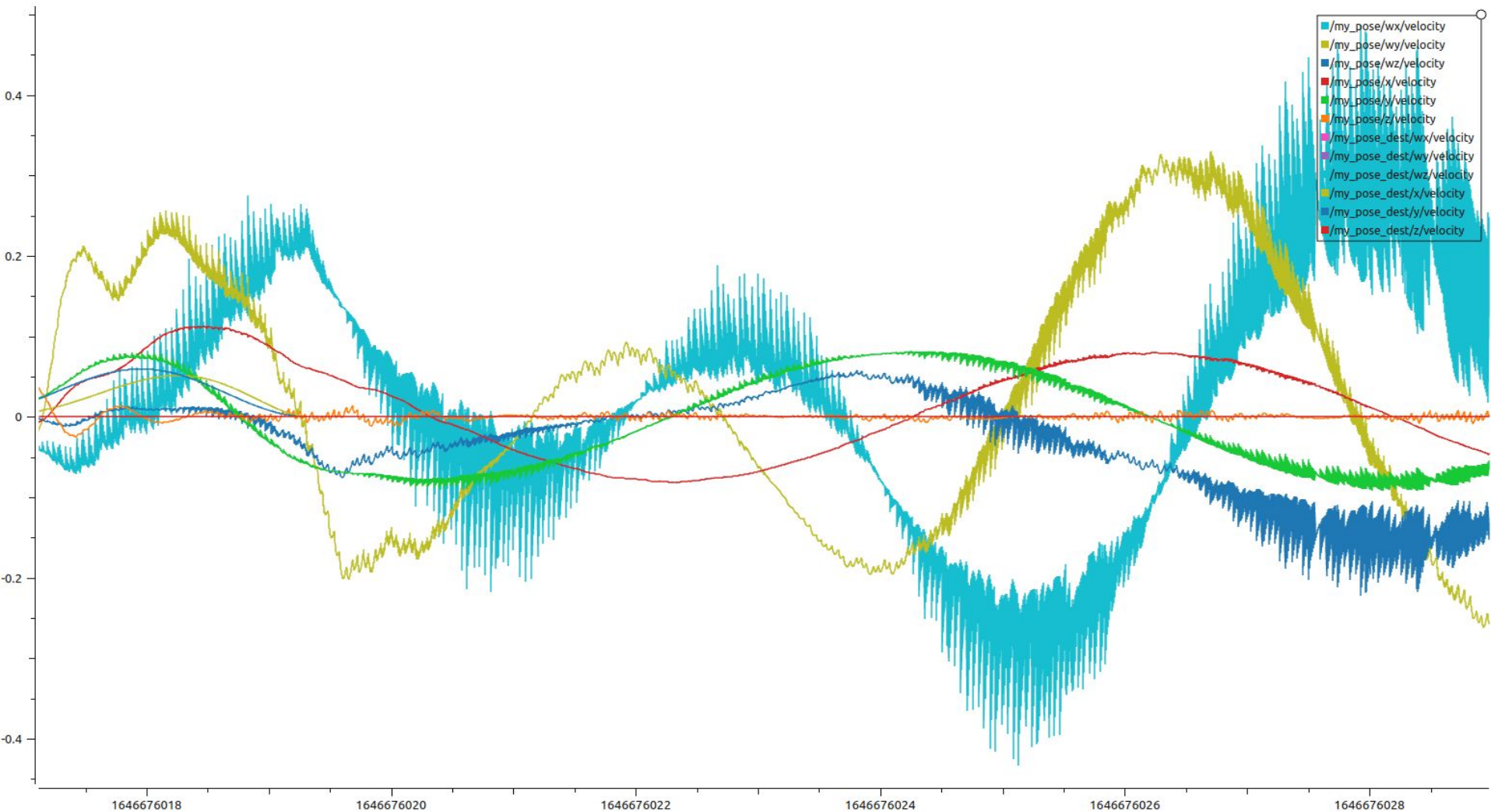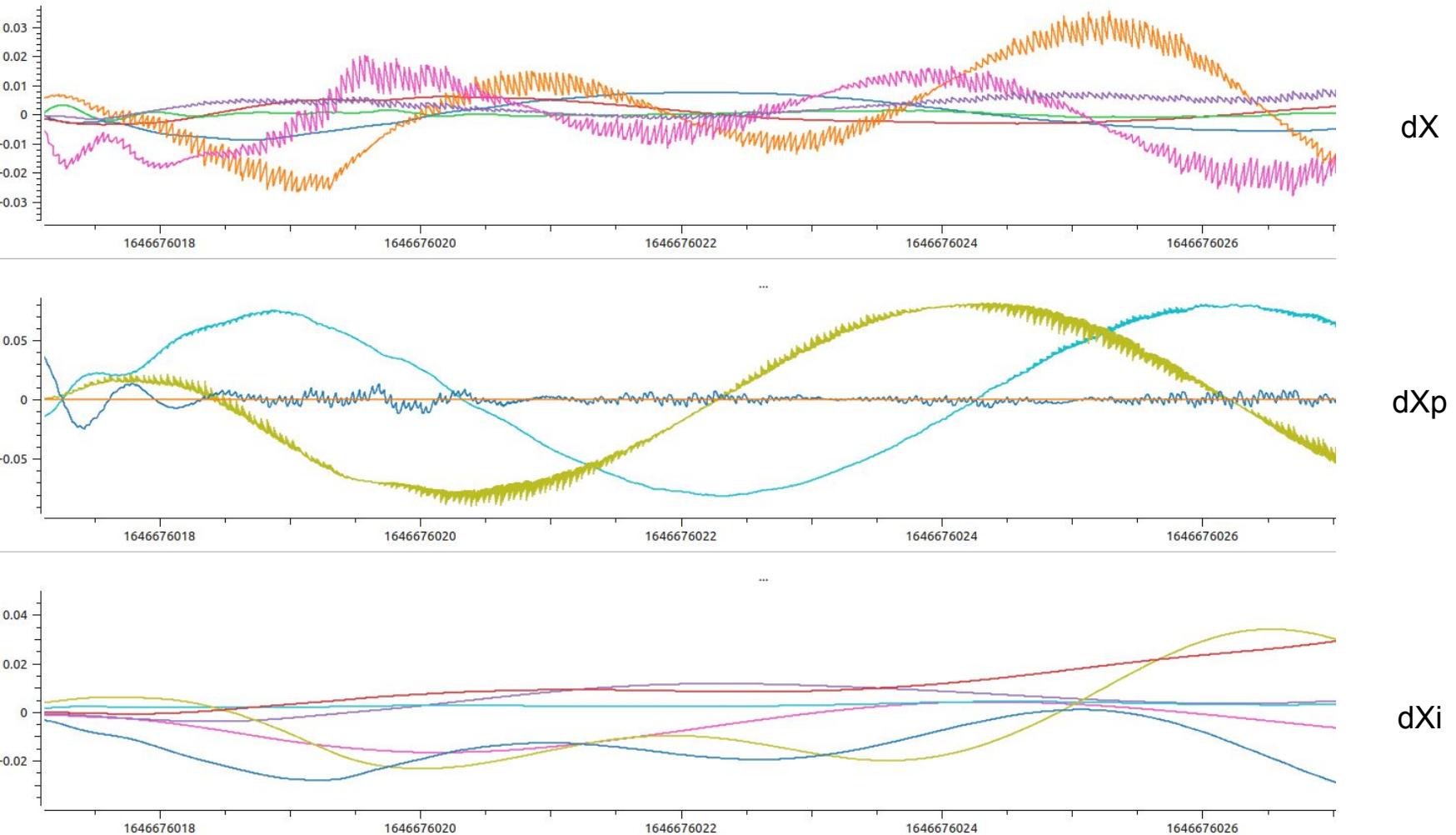
Joint Control Error Plots

dX

dXp

dXi

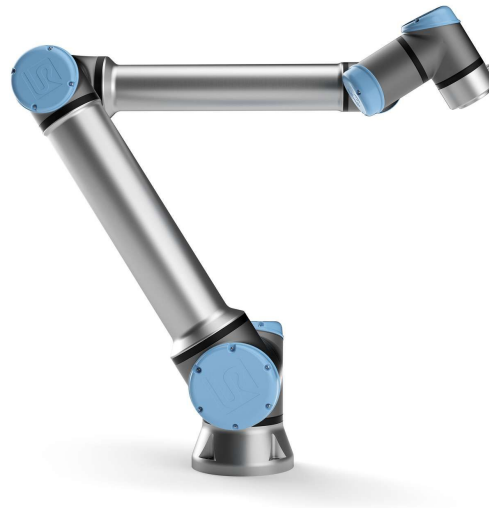# Operational Control Pose Plots

Operational Control Pose Velocity Plots

Operational Control Error Plots

Prof. Gordon Cheng (TUM)

Possible improvements

- Tune control parameters.
- Add impedance control with respect to joint 3.
- Improve switching between impedance on operational control.
- Divide the ros package in separate packages for different controls and visualization.
- Apply a low-pass filter to the inputs.

Thank you for your attention!